Enabling Global Multimedia Distributed Services based on Hierarchical DHT Overlay Networks

Isaias Martinez-Yelmo[1], Alex Bikfalvi[2], Carmen Guerrero[1], Ruben Cuevas[1], Andreas Mauthe[3]

[1]Universidad Carlos III de Madrid
Av. Universidad 30. 28911 Leganés. Madrid (Spain)
{imyelmo,guerrero,rcuevas}@it.uc3m.es

[2]IMDEA Networks Av. Mediterráneo 22. 28912 Leganés. Madrid (Spain)
alex.bikfalvi@imdea.org

[3]Computing Department at InfoLab 21
Lancaster University. Lancaster LA1 4WA (UK)
andreas@comp.lancs.ac.uk

Abstract

The provision of innovating multimedia services is a high priority for service providers. Due to the the high traffic volume characteristics of multimedia content, decentralised services are a desired solution for multimedia scenarios.

Based on the ongoing work of the IETF P2PSIP WG this paper defines a simple way to interconnect different domains using on peer-to-peer technology. The objective is to provide connectivity between users of different domains and allow decentralised multimedia services to be placed where they can be provided most effectively. Furthermore, the routing performance and routing state is analysed for the proposed hierarchical DHT overlay architecture.

1. Introduction

Nowadays, the provisioning of multimedia services (VoIP, VoD, Video Streaming, IPTV, etc) is one of the most important objectives of ISPs in delivering new and attractive services. However, due to their demanding requirements, these multimedia services have not been as widely deployed as would be expected. The success of applications such as Skype[1] [1], [14] is due to their P2P decentralised design despite their relative complexity. The problem with these applications is that they are closed privative solutions and their behaviour is unpredictable. Thus, a standardised decentralised scalable solution based on P2P overlay networks is desirable to facilitate a more large scale deployment of distributed multimedia services.

Although a number of solutions have been proposed to support decentralised multimedia services, the new approach of the IETF P2PSIP[2] Working Group is developing into a reference framework. P2PSIP [3] works on a peer-to-peer overlay based solution that facilitates a decentralised architecture. The developed protocol should be flexible enough [7] to support most of peer-to-peer networks. The idea is to provide a protocol that allows the implementation of any DHT overlay network such as Kademlia [11] or Chord [16]. However, the design of this protocol does not consider the inter-operation between different domains required to provide global multimedia services. In this paper we propose a way of solving this problem. Further, the routing performance and maintenance state of the different peers is assessed in order to show the performance of certain key-aspects.

Fig.1 illustrates our approach. The basic idea is that the different domains can deploy their own overlay network but global connectivity between them is established through a dedicated *interconnection overlay*. In this interconnection overlay each domain is represented through at least one *super-peer*. If an item, service or reference, is not in the same domain, a regular peer can asks its super-peer to route the query to the domain of the target peer. To support the

[1] http://www.skype.com

[2] http://www.p2psip.org

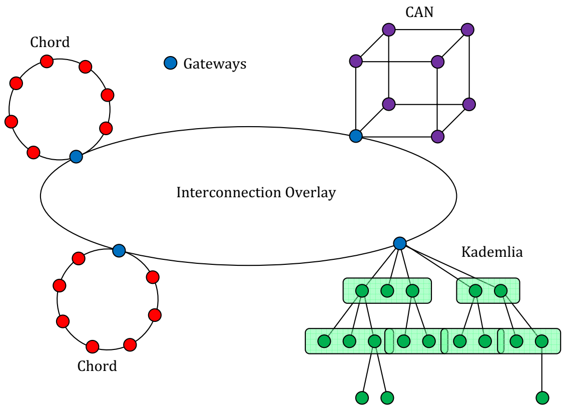

Figure 1. Hierarchical Overlay Architecture

routing between the different domains and the interconnection overlay, an extended identifier is used, which is formed by a *prefix ID* for the routing in the interconnection and a *suffix ID* for the routing in each domain.

Some advantages of this architecture are the network isolation and the improved scalability, which is intrinsic to the hierarchical architectures. However, issues such as a potential super-peer overload [2] have to be considered. This potential problem is limited to these peers since the routing state is not increased in legacy peers. For this reason, super peers can be dedicated entities allowing others (such as power-limited handheld devices, e.g. mobile phones) to be efficiently implemented.

We must highlight that in P2PSIP the overlay networks are used to retrieve the information about users and services, which is distributed across all peers in the overlay. Usually,this is location information (stored as IP address and transport port) and once this information is obtained, the negotiation of the service parameters is done using any suitable signalling protocol. Nevertheless, for compatibility reasons this protocol should be SIP [13].

The paper is structured as follows. Section 2 outlines the proposed hierarchical architecture. An analysis of its routing performance and the advantages of this approach are discussed in section 3. The particular case of a hierarchical Kademlia overlay network is studied in section 4. The theoretical is validated using the PeerFactSim.Kom framework. Finally, the related work is presented in section 5 and the conclusions and future work are presented in section 6.

2. Hierarchical DHT Overlay Networks

2.1. Hierarchical Space Domain of IDs

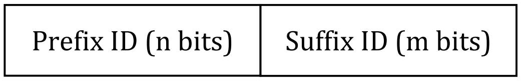

Figure 2. Hierarchical Space of Identifiers

In order to provide a hierarchical architecture for interconnecting different domains and assuming that the main goal of P2PSIP WG is to develop a framework to support any kind of DHT overlay network, we define a hierarchical space domain of identifiers. Thus, the *Hierarchical ID* (see Fig.2) is composed by two sub-identifiers: a *prefix ID* and a *suffix ID*. The prefix ID is used for the routing in the interconnection overlay between the different domains, whereas the suffix ID is used for routing queries only inside the own domain of a peer. This approach can be easily included in the P2PSIP protocol[7] because each header must contain an *overlay ID* which can be used as *prefix ID* and a *node ID* which can be used as *suffix ID*.

2.2. Service Mapping into the Hierarchical ID Space Domain

One of the main problems in a decentralised architecture is the mapping between the available information and/or services and the peers on the system. For example, if we consider a multimedia environment where services and users are identified by URI's, a resource is identified by `resource@example.com`. Now, the problem is to map these URI's to the proposed hierarchical ID space domain. In order to do this, the prefix ID is obtained by applying a hash to the domain of the URI: `PrefixID=hash(example.com)`

The suffix ID is generated as the hash of the URI: `SuffixID=hash(resource@example.com).`

Once the mapping between the URI and the Hierarchical ID has been established, a resource tuple containing the resource hierarchical ID, the URI and the resource itself, is stored at the peer with the closest peer ID. Depending on the DHT protocol this tuple can be also replicated in other peers.

2.3. Hierarchical DHT Overlay Network Operation

The behaviour of a hierarchical overlay network can be divided in two cases. The first one is when the search of a resource is bounded to the domain where a peer is attached. This case is simple since the search for resources is done inside the domain using a flat peer-to-peer overlay and where the routing is based only on the suffix ID. This situation is identified when the prefix ID of the resource is equal to the hash of the domain name.

When a resource in another domain has to be retrieved, the operation is more complex. For instance, this case could correspond to a VoIP call to a user in a different domain. In this circumstance, it is necessary to obtain the contact information published in the domain where the called party is registered. In the first stage, the initiator sends the search to its own super-peer. Super-peers are selected according to certain characteristics [12] and the selection mechanism can be integrated in the maintenance protocol of each DHT.

Since all the peers in a domain know their super-peer, they can send a query to the super-peer in one hop. When the super-peer receives the query, it routes the query through the interconnection overlay using the prefix ID. This query will arrive at the super-peer belonging to the domain matching the prefix ID from the query. From this point onwards, the query is forwarded inside the destination domain. If the query reaches a peer that has the desired resource, this peer replies this information to the initiator. This is possible because the query includes an {IP address, port} tuple where the requester is waiting the response. This answer must be compliant with the ongoing design of the P2PSIP protocol [7] that it is being defined by the IETF P2PSIP WG.

2.4. Main Characteristics of the proposed Hierarchical DHT Architecture

Our proposal has several advantages. First, the operations or primitives of the used DHT are not changed. Only some modifications are needed in the maintenance operations to include the selection and update of super-peers [12]. Furthermore, the routing state in peers does not increase because the number of peers that are maintained is the same.

If a global domain would be used to provide global connectivity, this domain would contain all peers in the different domains. If each domain has M peers and K domains want to obtain global connectivity, the number of peers is $N = M \cdot K$.In many DHT-based overlay networks the routing state has a logarithmic dependency with the number of peers. Therefore, we have $O(\log_B N)$ because of the logarithmic property: $O(\log_B M) + O(\log_B K) = O(\log_B(M \cdot K))$. This routing state applies only to super-peers, while regular peers only have to maintain the state of their own domain, which is only $O(\log_B M)$.

The drawback of this approach is the higher load needed to be supported by the super-peers [2] although this load is smaller than in other hierarchical DHT proposals [5], [6], [17], [18].

3. Routing Performance in a Hierarchical DHT Overlay Network

This section presents the routing performance in a system based on hierarchical DHT overlay network with the proposed *hierarchical ID*. This analysis is an extension of the work in [10] and a more elaborated model is presented.

We define the following notations:

- K: The number of domains.
- M_i: The number of peers in a domain i.
- N: All the peers from all the domains. In our case, it is considered that a peer cannot be attached to multiple domains, hence $N = \sum_{i=1}^{K} M_i$.
- ρ_{ij}: The probability of launching a query from the domain i to the domain j.
- $C(x)$: The number of hops needed to find a super-peer in the interconnection overlay depending on the number of super-peers x. This value depends on the type of overlay used in the interconnection overlay.
- $D_i(x)$: The number of hops needed to find a peer in a flat overlay of type i as function of the number of peers x belonging to the domain.

It is assumed that all the peers in a domain know their super-peer from the interconnection overlay. This assumption implies that only *one hop* is needed to reach the super-peer.

Taking into account the above definitions, we obtain the routing performance (RP) of this DHT-based hierarchical overlay networks. First of all, we define the cost of finding a peer in each overlay:

- $D_i\left(M_i\right)$: The cost of finding a peer in its own domain.
- $C\left(K\right)$: The cost of finding a super-peer in the interconnection overlay.

If the probability of obtaining an item in a domain from its super-peer is considered negligible and because the average number of peers in a domain is N/K with $N \gg K$, the average routing performance experienced by a peer in domain i can be written as follows:

$$RP_i = \rho_{ii} \cdot D_i\left(M_i\right) + \sum_{j=1, j\neq i}^{K} \rho_{ij} \cdot \left[1 + D_j\left(M_j\right) + C\left(K\right)\right] \tag{1}$$

The first term of the sum is the cost of searching something in the own domain of a peer, whereas the second term is the cost for the searches in the other domains.

The average number of hops is given by the next expression:

$$RP = \frac{1}{N} \cdot \sum_{i=1}^{K} M_i \cdot RP_i \tag{2}$$

If the number of peers is the same in all domains, we have:

$$RP = \frac{1}{K} \cdot \sum_{i=1}^{K} \cdot RP_i \tag{3}$$

Because we assumed that the number of peers is equal in all domains and each look-up in the overlay is considered randomly independent, we obtain that the probability of looking for a peer attached to other domain is equally distributed between all the foreign domains. In addition, the probability of looking for a peer in the own domain is different from the one of looking for a peer in other domains. Thus, the inter-domain query probability is $\rho_{ij} = \frac{1-\rho_{ii}}{K-1}$ and we can express Eq.1 as follows:

$$RP_i = \rho_{ii} \cdot D_i\left(M\right) + \sum_{j=1, j\neq i}^{K} \cdot \frac{1-\rho_{ii}}{K-1} \cdot \left[1 + D_j\left(M\right) + C\left(K\right)\right] \tag{4}$$

This relation is useful for some type of scenarios like VoIP in community networks where $\rho_{ii} > \rho_{ij}$ which implies that calls between peers of the same community are more frequent. For other services where the look-up probability in the own domain is the same as for foreign domains ($\rho_{ii} = \rho_{ij} = \frac{1}{K}$), we get Eq.5 (which is a simplified version of Eq.4):

$$RP_i = \frac{1}{K} \cdot D_i\left(M\right) + \sum_{j=1, j\neq i}^{K} \cdot \frac{1}{K} \cdot \left[1 + D_j\left(M\right) + C\left(K\right)\right] \tag{5}$$

Finally, if the same overlay is used in all domains the sum can be eliminated from the above expression:

$$\begin{aligned} RP_i = \frac{1}{K} \cdot D\left(M\right) + \frac{K-1}{K} \cdot \left[1 + D\left(M\right) + C\left(K\right)\right] = \\ = D\left(M\right) + \frac{K-1}{K} \cdot \left[1 + C\left(K\right)\right] \end{aligned} \tag{6}$$

From Eq.6 and Eq.3 we obtain the next equality:

$$RP = RP_i = D\left(M\right) + \frac{K-1}{K} \cdot \left[1 + C\left(K\right)\right] \tag{7}$$

4. Case Study: A Hierarchical Kademlia Overlay Network

In this section, we study the routing performance and the routing state in the case when a Kademlia overlay [11] is used in all domains and in the interconnection overlay. We selected Kademlia because it is one of the most used DHT-based overlays in peer-to-peer applications like e-Mule, Bittorent, etc.

Kademlia is an overlay network that has a routing performance and a routing state with a logarithmic dependency on the number of peers from the overlay, due to its XOR distance-based routing algorithm.

4.1. Analytical Analysis

In order to verify the efficiency of our solution, when the Kademlia protocol is used, we use the next equality: $C(x) = D(x) \sim \log_B x + c$, where B is a configuration parameter that allows to adjust the trade-off between the routing performance and the routing state in the peers.

We are going to substitute this expression in Eq.7, because the validation is performed with assumptions similar to the conditions that are valid for this expression. Therefore:

$$RP = RP_i \sim \log_B\left(M\right) + c + \frac{K-1}{K} \cdot \left[1 + \log_B\left(K\right) + c\right] \tag{8}$$

If $K \gg 1$ and taking into account the properties of the logarithm, we can write:

$$RP = RP_i \sim 1 + \log_B\left(M \cdot K\right) + 2c \tag{9}$$

For the routing state, the number of entries depends on the number of peers and on the setup parameter B. Actually, the number of overlay routing entries depends on $O(\log_B n)$ where n is the number of peers in the overlay. Super-peers have to support additional entries for the interconnection overlay. The total number of routing entries for a super-peer is approximately to $O(\log_B\left(K \cdot M\right))$.

If a flat overlay is used to connect all peers in different domains, peers would need $O(\log_B(K \cdot M))$ routing entries, but using the hierarchical architecture, legacy peers only need $O(\log_B M)$. Therefore, the routing state savings are significant if many domains are interconnected.

4.2. Validation via Simulation

In this section, we present several experimental results with the goal to assess the performance of a hierarchical Kademlia overlay network. The results have been obtained with a prototype implementation of the protocol and using the PeerfactSim.KOM[3] P2P network simulator [4]. The evaluation focused on determining the routing performance and the routing state (the average number of entries in the routing tables) with respect to the theoretical limits given for the canonical (or flat) overlay.

[3] http://peerfact.kom.e-technik.tu-darmstadt.de/

For the simulation results the corresponding 95% confidence intervals have been calculated. These confidence intervals give an error interval less than 5% which assures the consistency of the data recollected with the simulations.

4.2.1 Simulation Scenarios

The simulation scenarios considered different values for the number of domains and number of peers in a domain. Due to the capacity limits of the simulator, we simulated 5, 10 and 20 domains, with a number of peers per domain between 10 and 1000.

For these simulations, we have not taken into account the churn rate, although it is important to consider it in some particular scenarios. This simplification is based on the fact that Skype has a low churn rate [14] and we are interested in relatively similar scenarios.

In relation with the setup of the Kademlia overlay, the protocol has been configured with $B = 2^b = 2$, $k = 20$ and $\alpha = 1$. Although Kademlia defines a mechanism that accelerates the look-up process by increasing the size of the routing tables and the number of b bits considered at each hop, the simulation models use $b = 1$ and thus the results obtained reflect the worst case for the routing performance. The reason of using $\alpha = 1$ is to facilitate the comparison with other overlays that cannot easily parallelize their look-up operations. The value of k is used for the size of the buckets and also for the number of replicas of each item inside the Kademlia overlay.

4.2.2 Routing Performance

The routing performance was calculated for both *node look-up* and *value look-up* operations. The former are the result of the maintenance operations (refresh of the routing tables) and are performed solely inside the domain or inside the interconnection overlay between super peers. The latter are modelled based on peer behaviour of searching for stored values and can span two different domains. In addition, since the value look-ups take advantage of key-value replication, we expect the value look-ups to have a better performance. These operations finish as soon as a key is found.

According to the analytical model, and considering the assumptions on the simulation, the RP is estimated using Eq.8 or Eq.9 and depending on the value of *K*.

Figure 3 illustrates the average number of hops for inter-domain value look-up operations versus the number of peers in a domain for 5, 10 and 20 domains. The dependency is logarithmic with the number of peers in a domain (linear on a logarithmic scale), when the number of super peers is kept constant. The difference between the values obtained for each number of domains represents the mean of the extra number of hops needed when the number of super-peers in the interconnection overlay increases. Because the increase is almost constant while the number of super peers doubles, it also proves the logarithmic dependency of the routing performance with the size of the interconnection overlay.

Figure 4 demonstrates the expected difference between value look-up and node look-up (Log2 stands for $\log_2$). Because in our scenarios, node look-up operations cannot terminate in another domain (as they are generated by intra-domain maintenance), only intra-domain routing performance data are considered. As expected, in average the value look-ups require fewer hops due to replication. The difference is higher when the number of peers inside the domain is comparable to the replication parameter ($k = 20$) and becomes negligible when the number of domain peers is large enough for the replication to have an important effect. The figure also contains the maximum number of hops that were obtained during the same simulation for a visual comparison with the theoretical logarithmic limit ($\log_2 M$).

Finally, we analyse the worst case routing behaviour. Figure 5 compares the maximum number of hops obtained for inter-domain routing for 5 and 10 domains with the theoretical values given by Eq.8 or Eq.9. The logarithmic dependency can be observed as expected.

4.2.3 Routing State

The evaluation of the routing state intends to determine whether the average number of routing entries maintained by the peers lay within the expected ranges and to illustrate the behaviour of the routing state when the number of domains changes. For this, we examine the routing tables used for routing inside domains.

Figure 6 shows the obtained results. We have that: $NE \in [\log_2 N, k \log_2 N]$, where NE is the average number of routing entries. In addition, we can observe a slight dependency between the number of domains and the value of the routing state. Since the routing state is determined solely by the interaction between peers, the explanation for this dependency is that the simulation scenarios use the same number of value look-up operations. In general, the value look-ups are originated in one domain and usually terminated in another domain. However, if the number of domains is small the number of operations that originate and terminate in the same domain increases and consequently the number of routing entries also increases according to the standard Kademlia protocol to populate the bucket entries. Node look-ups cannot influence the routing state because they take place only inside a domain and have no relationship to the number of domains.

The number of routing entries for super-peers is quite similar to the values obtained for the intra-domain routing tables.

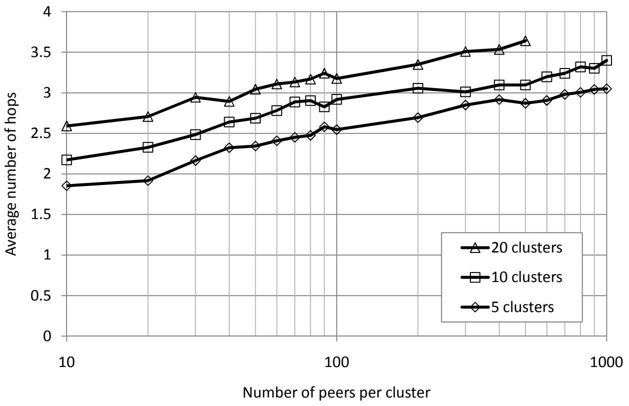

Figure 3. The routing performance for value look-up operations

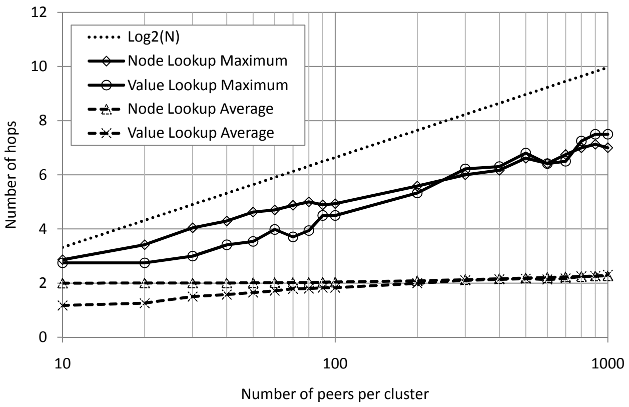

Figure 4. The routing performance for intra domain value look-up and node look-up operations

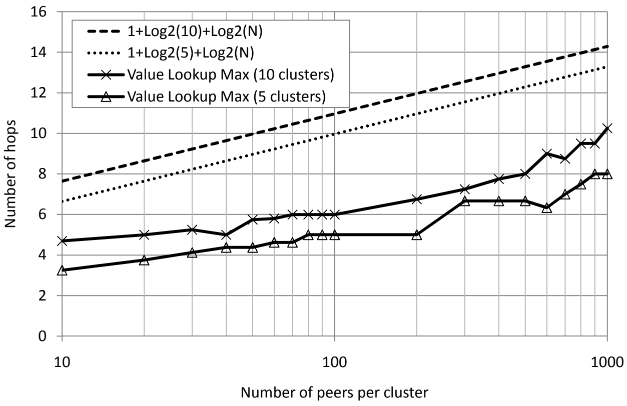

Figure 5. The worst case routing performance for value look-up operations

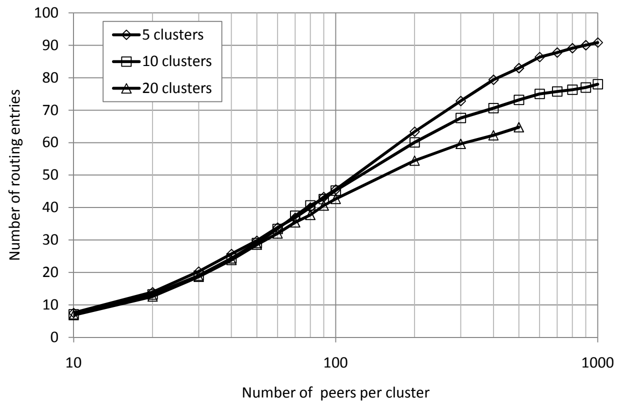

Figure 6. Routing state for intra-domain routing tables

5. Related Work on Hierarchical Overlay Networks

Overlay networks usually require $O(\log_B N)$ peer hops to reach the desired destination and $O(\log_B N)$ routing entries to maintain the desired structure. This complexity ensures good scalability but it is desirable to have further improvements. Thus, hierarchical overlay networks are being proposed because its benefits are clear [9].

The first approach is to delegate all the work to superpeers [6], [18]. They maintain the overlay network and perform all the necessary actions and legacy peers only have to register their information to their peers. Other studies are focused on optimising some parameter like the delay. In [17] a low delay hierarchical overlay network based on Chord is proposed. The drawback is the high routing state needed because *all* the peers in the overlay are attached to *all* the levels in a *n-level* hierarchy. A less aggressive design with the same objective is presented in [5] but the hierarchy is built with the constraint of limiting the maintenance cost to the flat counterpart.

The main problem is the selection of super-peers. This selection can be based on the computation capacity of a peer, the available bandwidth to receive and process the queries and the session time to assure a stable set of superpeers [12]. Furthermore some mechanism must be provided to distribute the super-peer related information. One option is to piggyback this information [8].

6. Conclusions and Future Work

The objective of the architecture proposed in this paper is to enable the interconnection of different domains in order to support global decentralised multimedia services.

Peers of the same domain are connected via a domain overlay network and by using the `SuffixID=hash(user@example.com)` to route queries. In order to get connectivity with other domains it is necessary to have at least one super-peer in each domain. An interconnection overlay is maintained between the super-peers where the routing is based on `PrefixID=hash(example.com)` values.

The average number of hops that a peer must perform in our architecture is given by Eq.1. However, based on certain justifiable assumptions, it is simplified to Eq.7. Furthermore, we consider that the routing state of "normal" peers does not change. In fact, only super-peers are exposed to a higher load. These super-peers might actually be dedicated entities employed by the domain for this specific task.

Finally, a simulation of hierarchical Kademlia overlay network has been performed. The routing performance is below the worse case expected through the analytical evaluation. The information related to a specific user or service can be retrieved in 2 to 3.5 hops for a number of domains between 5 and 20 and with a number of peers per domain between 10 and 1000. These results illustrate the scalability of the solution.

In the next step of our research our main goal is to take into account the churn rate based on results presented in [15].

Acknowledgement

This work has been supported by the European Union under the IST Content NoE [4] (FP6-2006-IST-038423), by the Regional Government of Madrid under the BioGrid-Net[5] project (CAM, S-0505/TIC-0101) and by the Ministry of Science and Innovation under the CONPARTE project (MEC, TEC2007-67966-C03-03/TCM).

References

[1] S. A. Baset and H. G. Schulzrinne. An analysis of the skype peer-to-peer internet telephony protocol. *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–11, April 2006.

[2] B. Beverly Yang and H. Garcia-Molina. Designing a super-peer network. In *Data Engineering, 2003. Proceedings. 19th International Conference on*, pages 49–60, 2003.

[3] D. Bryan, P. Matthews, E. Shim, and D. Willis. Concepts and terminology for peer to peer sip, November 2007. Internet Draft draft-ietf-p2psip-concepts-01.txt.

[4] V. Darlagiannis, A. Mauthe, N. Liebau, and R. Steinmetz. An Adaptable, Role-based Simulator for P2P Networks. *Proceedings of the International Conference on Modeling, Simulation and Visualization Methods*, pages 52–59, 2004.

[5] P. Ganesan, K. Gummadi, and H. Garcia-Molina. Canon in g major: designing dhts with hierarchical structure. In *Distributed Computing Systems, 2004. Proceedings. 24th International Conference on*, pages 263–272, 2004.

[6] L. Garces-Erice, E. W. Biersack, K. W. Ross, P. A. Felber, and G. Urvoy-Keller. Hierarchical p2p systems. In *Proceedings of ACM/IFIP International Conference on Parallel and Distributed Computing (Euro-Par)*, 2003.

[7] C. Jennings, B. Lowekamp, E. Rescorla, J. Rosenberg, S. Baset, and H. Schulzrinne. Resource location and discovery (reload), June 2008. Internet Draft draft-bryan-p2psip-reload-04.txt.

[8] Y.-J. Joung and J.-C. Wang. Chord2: A two-layer chord for reducing maintenance overhead via heterogeneity. *Computer Networks*, 51(3):712–731, Feb 2007.

[9] M. Kwon and S. Fahmy. Synergy: an overlay internetworking architecture. In *Computer Communications and Networks, 2005. ICCCN 2005. Proceedings. 14th International Conference on*, pages 401–406, 2005.

[10] I. Martinez-Yelmo, R. Cuevas, C. Guerrero, and A. Mauthe. Routing performance in hierarchical dht-based overlay networks. In *In Proceedings on 16th Euromicro International Conference on Parallel, Distributed and network-based Processing*, Feb. 2008.

[11] P. Maymounkov and D. Mazieres. *IPTPS 2002 Cambridge, MA, USA, March 7-8, 2002. Revised Papers*, volume 2429/2002 of *Lecture Notes in Computer Science*, chapter Kademlia: A peer-to-peer information system based on the XOR metric, pages 53–65. Springer, 2002.

[12] S.-H. Min, J. Holliday, and D.-S. Cho. Optimal super-peer selection for large-scale p2p system. In *Hybrid Information Technology, 2006. ICHIT'06. Vol 2. International Conference on*, volume 2, pages 588–593, 2006.

[13] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol. RFC 3261 (Proposed Standard), June 2002. Updated by RFCs 3265, 3853, 4320, 4916.

[14] D. Rossi, M. Melia, and M. Meo. A detailed measurment of skype network traffic. In *IPTPS 2008*, 2008.

[15] M. Steiner, T. En-Najjary, and E. W. Biersack. A global view of kad. In *IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 117–122, New York, NY, USA, 2007. ACM.

[16] I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications. *IEEE/ACM TRANSACTIONS ON NETWORKING*, 11(1), 2003.

[17] Z. Xu, R. Min, and Y. Hu. Hieras: a dht based hierarchical p2p routing algorithm. In *Parallel Processing, 2003. Proceedings. 2003 International Conference on*, 2003.

[18] S. Zoels, Z. Despotovic, and W. Kellerer. Cost-based analysis of hierarchical dht design. In *Peer-to-Peer Computing, 2006. P2P 2006. Sixth IEEE International Conference on*, pages 233–239, 2006.

[4] http://www.ist-content.eu

[5] http://www.biogridnet.org